Attorney Docket No.: 010327-008800US
Client Reference No.: 35

PATENT APPLICATION

METHOD AND SYSTEM FOR RECOVERING AND ALIGNING SYNCHRONOUS DATA OF MULTIPLE PHASE-MISALIGNED GROUPS OF BITS INTO A SINGLE SYNCHRONOUS WIDE BUS

Inventor(s): Ran Sandrovitz, a citizen of Israel, residing at
3965 Derryfield Ct.,
Dublin, CA 94568

Assignee: Network Equipment Technologies, Inc.
6900 Paseo Padre Parkway
Fremont, CA 94555

Entity: Large Business Concern

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, California 94111-3834
Tel: 415-576-0200

Attorney Docket No.: 010327-008800US
Client Reference No.: 35

METHOD AND SYSTEM FOR RECOVERING AND ALIGNING SYNCHRONOUS DATA OF MULTIPLE PHASE-MISALIGNED GROUPS OF BITS INTO A SINGLE SYNCHRONOUS WIDE BUS

BACKGROUND OF THE INVENTION

[0001] The present invention generally relates to data communications and, more specifically, to a method and system for recovering and aligning data transmissions over a synchronous bus.

[0002] In electronic equipment, various internal components communicate and exchange information and data with each other. Such communications are usually conducted over a media, such as a backplane, traces, wires or cables. A transmitter or source of traffic, for example, a card transmitting data on a 32-bit wide synchronous bus and a SOC (start-of-cell) signal onto a media, transmits data toward a receiver which is located at the other end of the media. The electrical signals going across the media are generally routed in groups, where each group of signals/bits is length-matched. This means that the propagation latency of traversing the media is the same for all signals/bits within the same group; however, different groups may have different latencies even though the different groups are sent from the transmitter at the same time. Different groups may arrive at the receiver in different times due to delta in media length or different delays across the media. As a result, different groups may be phase-misaligned or skewed.

[0003] For example, assume that there are eight (8) groups of four (4) bits each, a source-synchronous clock signal ("CLK") and an additional SOC signal, all of which form a wide synchronous bus. All thirty-two (32) bits (plus the SOC signal) of the bus are fully synchronous and aligned as they exit the transmitter. However, due to the latency mentioned above, not all eight (8) groups (32 bits) may arrive at the receiver at the same time.

[0004] Data is only meaningful if the original thirty-two (32) bits can be identified and assembled. In other words, all eight (8) groups need to be de-skewed at the receiver in order to recover the original thirty-two (32) bits. To capture the original thirty-two (32) bits at the receiver, the CLK can be used. Using the CLK, however, would result in poor performance. That is because the CLK would have to run at low frequency to ensure setup/hold times are not violated in the presence of various delays of the different groups across the media.

[0005] Hence, it would be desirable to provide a method and system that is capable of resolving the foregoing problem, as well as others, by recovering and aligning data transmissions over a synchronous bus in an efficient manner.

BRIEF SUMMARY OF THE INVENTION

[0006] A system for recovering and aligning synchronous data transmissions is disclosed. In one embodiment, the system includes a transmitter configured to transmit a source clock signal and a number of data groups over a number of channels. The data groups are transmitted during the same clock cycle pursuant to the source clock signal. Each data group is transmitted over a corresponding channel.

[0007] The system also includes a receiver configured to receive the source clock signal and the data groups over the corresponding channels. The receiver further includes: for each channel, (a) a local clock configured to generate a local clock signal based on the source clock signal, the local clock signal being phase-shifted from the source clock signal by a predetermined amount of phase shift, (b) a logic device configured to clock in the data group received over the channel using the local clock signal, (c) a sequence number generator configured to generate a sequence number associated with the data group, (d) a FIFO configured to store and output the clocked-in data group and the associated sequence number, (e) a memory device configured to store the clocked-in data group from the FIFO using the associated sequence number as a memory address, the memory device further configured to output a predetermined portion of its contents after a predetermined capacity threshold or level is reached.

[0008] In one embodiment, the transmitter is further configured to transmit a start-of-cell signal to the receiver. Upon the receiver detecting the start-of-cell signal having a specific value for a predetermined cycle period, the sequence number generators are synchronized.

[0009] In one implementation, the receiver is implemented using a number of field programmable gate arrays and the local clocks are implemented using digital clock managers associated with the field programmable gate arrays.

[0010] Reference to the remaining portions of the specification, including the drawings and claims, will realize other features and advantages of the present invention. Further features and advantages of the present invention, as well as the structure and operation of various embodiments of the present invention, are described in detail below with respect to accompanying drawings, like reference numbers indicate identical or functionally similar elements.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011] FIG. 1 is a simplified schematic block diagram illustrating an exemplary embodiment of the present invention; and

[0012] FIG. 2 is a simplified schematic block diagram further illustrating one embodiment of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

[0013] The present invention in the form of one or more exemplary embodiments will now be described. In an exemplary aspect, a method is disclosed for de-skewing groups of bits received at a receiver to recover the full-width bus in a common clock domain using a source-generated roll-over counter and a dual-port random access memory at the receiver.

[0014] FIG. 1 is a simplified schematic block diagram illustrating an exemplary embodiment of the present invention. In this embodiment, the system 10 includes a transmitter 12, a media 28, a data recovery/alignment ("DR/A") module 14 and a data processing block 16. The transmitter 12 can be any type of device that is capable of transmitting traffic including, for example, a signal card. The transmitter 12 transmits a number of signals over the 28 media having a number of channels to the DR/A module 14. The signals include a SOC signal 18, data groups 22a-h, and a clock (CLK) signal 20.

[0015] In this embodiment, each data group 22 is made up of four (4) bits. The data groups 22a-h respectively represent eight (8) groups and collectively make up a 32-bit word. It should be noted that each of the data groups 22a-h may have a different delay or latency going from the transmitter 12 to the DR/A module 14 through the media 28. However, within each data group 22, all four (4) bits have the same delay, i.e., all four (4) bits are length-matched and arrive at the DR/A module 14 at the same time.

[0016] The SOC signal 18 is used to indicate the start of a cell. The CLK signal 20 is used by the transmitter 12 to control transmission of the data groups 22a-h for each cycle. In this embodiment, a 32-bit word is transmitted by the transmitter 12 every cycle based on the CLK signal 20. A cell contains a fixed amount of data. If a cell contains fifty-two (52) bytes, then the SOC signal 18 is sent every thirteen (13) cycles (four (4) bytes (32 bits) are sent every cycle).

[0017] In an exemplary aspect, the DR/A module 14 performs two functions to recover the data groups 22a-h transmitted from the transmitter 12, more specifically, a data recovery function and a word alignment function, as will be further described below. The DR/A module 14 includes a number of DR/A sub-modules 14a. In one implementation, the DR/A module 14 includes a number of logic devices, such as, field programmable gate arrays (FPGAs).

[0018] FIG. 2 is a simplified schematic block diagram further illustrating one embodiment of the DR/A sub-module 14a operating on one of the data groups 22a-h, for example, data group 22a. As noted above, data group 22a is made up of four (4) bits. It should be noted that other DR/A sub-modules 14a operate on the other data groups 22b-h in a similar manner.

[0019] To carry out the data recovery function, a private source-synchronous or phase-shifted clock is created locally for data group 22a. It should be noted that all eight (8) data groups 22a-h arriving from the media 28 are only frequency-locked to the CLK signal 20 but not phase-aligned. Therefore, in order to clock in the data groups 22a-h, a phase-shifted version of the CLK signal 20 is created for each data group 22. A phase-shifted version of the CLK signal 20 is also created for the SOC signal 18. In this embodiment, there are then nine (9) "phase domains" or "phase-shifted" clocks. The foregoing allows each data group 22 to have the best timing margin, thus, allowing the CLK signal 20 to have high frequency which, in turn, increases the bandwidth of the transmission across the media 28.

[0020] In one implementation, the private phase shift per data group 22 is done using the digital clock managers (DCMs) in the FPGAs. An example of such FPGAs is the Spartan3 FPGA manufactured by Xilinx. A DCM can do fine phase shift of the CLK signal 20 in granularity of, for example, 50ps, such that all four (4) bits of each data group 22 are clocked in the center of the eye opening.

[0021] The amount of phase shift to be applied to the CLK signal 20 to create the phase-shifted clock for each data group 22 varies depending on the latency delay of each data group 22 with respect to the CLK signal 20. The latency delay of each data group 22 and the CLK signal 20 can be determined empirically based on measurements or calculated based on known parameters. When the latency delay of a data group 22 is known with respect to that of the CLK signal 20, the private phase shift for that corresponding data group 22 can be determined so that such data group 22 can be clocked correctly using its associated phase-shifted clock.

[0022] As mentioned above, the DR/A sub-module 14a also performs a word alignment function, described as follows. Data for each data group 22 and their associated sequence numbers are stored into an associated FIFO (First-In-First-Out) memory 24 that is clocked in using its associated phase-shifted clock. As will be further described below, the function of FIFO memory 24 is to cross clock domains from the local/private phase-shifted clock to the common clock domain. In this illustration, four (4) bits are stored into the FIFO memory 24 every clock cycle under the control of the associated phase-shifted clock. In addition, a corresponding sequence number is concatenated with the four (4) bits and stored into the memory 24. Sequence numbers, as will be further explained below, are tags respectively applied per phase-shifted clock cycles to the data of the eight (8) data groups 22a-h that were originally sent together by the transmitter 12 per clock cycle based on the CLK signal 20 so as to allow them to be later identified as having been transmitted during the same clock cycle. The sequence number for each 4-bit group is incremented every clock cycle. The current sequence number is maintained by a sequence number counter. In essence, data from each data group 22 is written into its associated FIFO memory 24 with sequence numbers.

[0023] Sequence numbers are used to respectively identify data cycles from all eight (8) data groups 22a-h that were originally sent together by the transmitter 12. All corresponding initial sequence numbers for data from the data groups 22a-h are initialized to a specific initial value. This initial value is provided by the transmitter 12 and forwarded to the DR/A module 14 via an alignment cell. From that moment on, the transmitter 12 and the DR/A module 14 independently increment a sequence number counter starting with this initial value. Each clock/data cycle is associated with and identified by a unique sequence number. The alignment cell is a cell (e.g., nine (9) cycles long) that is signified by a special SOC signal 18 having a value of "1" for four (4) consecutive clock cycles (or some other predefined cycle-count). In one embodiment, the alignment cell includes seven (7) cycles of pre-defined data pattern and two (2) cycles of data representing the initial value.

[0024] When the special SOC signal 18 is detected by the DR/A module 14, the relevant FPGAs associated with the data groups 22a-h are informed about this event. Since the SOC signal 18 is not in phase with the data groups 22a-h, the SOC signal 18 is treated as asynchronous. When the relevant FPGAs associated with the data groups 22a-h receive this indication, they look for the predefined data pattern embedded in the alignment cell and lock to it. The sequence number value embedded in the alignment cell is then retrieved and the sequence number counters associated with the corresponding data groups 22a-h are then set to this value thereby synchronizing the sequence numbers in all DR/A sub-modules 14a of all the data groups 22a-h. It should be noted that alignment cells having specific values can be sent periodically by the transmitter 12 to align any newly plugged in card(s), assuming the media 28 is shared between multiple receiver cards, and correct any possible misalignment caused by conditions, such as, electro-static discharge. For example, since the transmitter 12 independently maintains a sequence number counter, the correct sequence number can be forwarded to the DR/A module 14 to provide the appropriate alignment.

[0025] For each FIFO memory 24 associated with a corresponding data group 22, memory contents are read out and written into a corresponding dual port random access memory (DPRAM) 26. More specifically, each 4-bit group is written into the DPRAM 26 using its associated sequence number as the "write" address. In other words, memory contents of the FIFO memories 24 associated with the data groups 22a-h are transferred over to a number of DPRAMs 26. It should be noted that the "read" operations performed on the FIFO memories 24 are controlled by a common clock domain which is common to all the FPGAs used to recover the data groups 22a-h. Reading the FIFO memory 24 is done in small bursts based on the "FIFO Half Full" indication to avoid FIFO underrun or overflow. By using sequence numbers as "write" addresses in the DPRAMs 26, data cycles of data groups 22a-h having the same sequence number are stored at the same locations in the DPRAMs 26. Data belonging to data groups 22a-h and having the same sequence number means they were originally sent out together by the transmitter 12 during the same clock cycle and, hence, should be treated together. Since the DPRAM 26 can be read or written to at the same time, special care is taken to ensure that no 'read' and 'write' operations are performed with the same address concurrently.

[0026] Once the DPRAM 26 is filled up to a predefined capacity threshold, data can be read out beginning with a specific starting address. It should be noted that there is a DPRAM 26 for each data group 22. Once one of the DPRAMs 26 has reached the predefined capacity threshold, 'read' operations are initiated for all DPRAMs 26 in a coordinated fashion, such that all DPRAMs 26 start sending a burst of data out on the same common clock cycle starting from the same DPRAM address. This is because all the DPRAMs 26 collectively make up the data groups 22a-h. In one embodiment, once the predefined capacity threshold is reached, the FPGA that handles the SOC and CLK signals 18 and 20 initiates a burst of reads from the DPRAMs 26. Data from the reads is forwarded to the data processing block 16. Once a predefined low watermark threshold is reached, the 'read' operations are discontinued. The predefined low watermark threshold can be, for example, a predetermined number of blocks that are to be read out from the DPRAM 26.

[0027] It should be noted that the DPRAMs 26 may be filled up at the same rate but at different phases (i.e., skewed), because the corresponding FIFO memory 24 providing the data to the DPRAM 26 is driven by the phase-shifted clock associated with a particular data group 22 and due to the different latency across the media 28. Due to such phase difference, an appropriate margin is built into the predefined capacity threshold to allow all the DPRAMs 26 to collect sufficient data for the subsequent 'read' operations. The appropriate margin can be determined based on the maximum latency variation of the data groups 22a-h traveling from the transmitter 12 to DR/A module 14.

[0028] Also, it should be noted that the common clock for reading the DPRAMs 26 is faster than the local/private phase-shifted clocks used for writing data to the FIFO memories 24.

[0029] In an exemplary implementation, the present invention is implemented using a combination of hardware and software in the form of control logic, in either an integrated or a modular manner. Based on the disclosure and teachings provided herein, a person of ordinary skill in the art will know of other ways and/or methods to implement the present invention.

[0030] It is understood that the examples and embodiments described herein are for illustrative purposes only and that various modifications or changes in light thereof will be suggested to persons skilled in the art and are to be included within the spirit and purview of this application and scope of the appended claims. All publications, patents, and patent applications cited herein are hereby incorporated by reference for all purposes in their entirety.